Unviersity of Virginia, Department of Computer Science
CS216 Spring 2006

PS6 Comments

1. The iload instruction is used to load a local integer, so any access to previously stored integer should produce the instruction iload.

```
public static void main(String[] args){
        int i=10;
        System.out.println(i);
}
```

2. The value of the first parameter is stored in location 0, so we just need to call a method that takes it twice:

```
public static void main(String[] args) {
        call(args,args);
}

private static void call(String[] a,String[] b)
{
}
```

The resulting class file excerpt is:

```
Method void main(java.lang.String[])
   0 aload_0
   1 aload_0
   2 invokestatic #2 <Method void call(java.lang.String[], java.lang.String[])>
   5 return
```

3. The invokevirtual instruction is used to call a method. To have 3 consecutive invokevirtual, we need to have three consecutive calls. Here's one way:

```
public class Dup {
   public Dup dip () {
      return this;
   }

   public Dup dup () {
      return this.dip ().dip ().dip ();
   }
   …
```

The code for the dup method is:

```
Method Dup dup()
  0 aload_0
  1 invokevirtual #2 <Method Dup dip()>
  4 invokevirtual #2 <Method Dup dip()>
  7 invokevirtual #2 <Method Dup dip()>
 10 areturn
```

4. The wide instruction is used to modify instructions to take a two-byte parameter. To see a wide iload instruction, we need to have more than 256 local variables:

```
int x1;
int x2;
...
int x257;
System.out.println(x257);
```

5. Changing Election.java to print out i=3 enables you to recognize the corresponding byte codes for the code you inserted after you run D-java and then modify the byte code to print the memory location of e. In order to find the memory location of e, you had to replace iload_3, which put the three on the stack, with aload_1, the location of e on the stack. When the call to print is made, your result should be a memory location such as 268668224 (the actual address may vary according to how memory is allocated on the particular virtual machine you used).

6. Your modifications should look something like the following:

```
ldc 268668224
astore_0
```

The constant after the ldc is the result you got for question 5. The ldc instruction, which pushes a number or a string onto the stack, to push the reference to e. After pushing the reference onto the stack, this stores it in a local variable (in this case local variable 0) in order to be able to access it easily. Note that this code violates type safety — we are using an integer value as an address and treating it as an object.

7. It was easiest to develop the code in Election.java so you could use the Java compiler to produce most of the bytecodes you need. For the Java compiler to compiler the code, you first had to (temporarily) change some of the private variables in ElectionResults class to public values. Doing this would allow you to lookup the Dog Catcher Office, obtain Sarge's record, and change the number of votes to zero using this code:

```
OfficeResult o = e.lookupOffice ("Dog Catcher");
CandidateRecord c = o.getRecord ("Sarge");
c.votes = 0;
```

After compiling and testing this code, you could use D-Java to obtain the corresponding bytecodes, and move them into CompleteElection.j. Then, undo the changes to ElectionResults since the final code will get around the visibility restrictions by circumventing Java's code safety mechanisms (in this case, but using -noverify).

You needed to run with -noverify because you loaded the election object e in number 6 onto the stack as an integer, and then stored it as an object; without the -noverify, running Election.class would produce a type safety error. The following code accomplishes this task:

```
.method public static displayAnimation()V
   .throws java/lang/Exception
   .limit stack 2
   .limit locals 3

   ldc 268668224
   astore_0
   aload_0
   ldc "Dog Catcher"
   invokevirtual ElectionResults/lookupOffice(Ljava/lang/String;)LOfficeResult;
   astore_1
   aload_1
   ldc "Sarge"
   invokevirtual OfficeResult/getRecord(Ljava/lang/String;)LCandidateRecord;
   astore_2
   aload_2
   iconst_0
   putfield CandidateRecord/votes I
   return
.end method
```

8-10. We discussed these questions in Class 21.